



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/762,866	01/22/2004	Kevin J. Turpin	4001-0121	5921
81877 7590 03/17/2011 ADVANTEDGE LAW GROUP, LLC 922 W. Baxter Dr. Suite 100 South Jordan, UT 84095				
EXAMINER TIMBLIN, ROBERT M				
ART UNIT		PAPER NUMBER		
2167				
NOTIFICATION DATE		DELIVERY MODE		
03/17/2011		ELECTRONIC		

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

81877@alg-ip.com



UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents  
United States Patent and Trademark Office  
P.O. Box 1450  
Alexandria, VA 22313-1450  
[www.uspto.gov](http://www.uspto.gov)

**BEFORE THE BOARD OF PATENT APPEALS  
AND INTERFERENCES**

Application Number: 10/762,866  
Filing Date: January 22, 2004  
Appellant(s): TURPIN ET AL.

\_\_\_\_\_  
Jonathan Lee (56,561)  
For Appellant

**EXAMINER'S ANSWER**

This is in response to the appeal brief filed 5/24/2010 appealing from the Office action mailed 3/1/2010.

**(1) Real Party in Interest**

The examiner has no comment on the statement, or lack of statement, identifying by name the real party in interest in the brief.

**(2) Related Appeals and Interferences**

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

**(3) Status of Claims**

The following is a list of claims that are rejected and pending in the application:

1-8, 10-18, 20-28, 30-38, and 40-43.

**(4) Status of Amendments After Final**

The examiner has no comment on the appellant's statement of the status of amendments after final rejection contained in the brief.

**(5) Summary of Claimed Subject Matter**

The examiner has no comment on the summary of claimed subject matter contained in the brief.

**(6) Grounds of Rejection to be Reviewed on Appeal**

The examiner has no comment on the appellant's statement of the grounds of rejection to be reviewed on appeal. Every ground of rejection set forth in the Office action from which the appeal is taken (as modified by any advisory actions) is being maintained by the examiner except for the grounds of rejection (if any) listed under the subheading "WITHDRAWN

REJECTIONS.” New grounds of rejection (if any) are provided under the subheading “NEW GROUNDS OF REJECTION.”

**(7) Claims Appendix**

The examiner has no comment on the copy of the appealed claims contained in the Appendix to the appellant’s brief.

**(8) Evidence Relied Upon**

6,615,365	Jenevein et al.	9-2003
6,026,016	Gafken	2-2000
GB Patent No. GB2376093	Leech, Guy	2002-12-04

**(9) Grounds of Rejection**

The following ground(s) of rejection are applicable to the appealed claims:

**Claim Rejections - 35 USC § 103**

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

**Claims 1-8, 11-18, 21-28, 31-38, and 41-43 are rejected under 35 U.S.C. 103(1) as being unpatentable over U.S. Patent 6,615,365 B1 issued to Jenevein et al. (‘Jenevein’ hereafter) in view of Gafken (U.S. Patent 6,026,016).**

**With Respect to claim 1.** Jenevein teaches A method for backing up a file system in a partition comprising a plurality of allocation units, the method comprising:

creating a locally-stored image file (step 704) by copying (col. 5 line 31-32) each allocation unit (col. 3 line 52-53, col. 5 line 31; e.g. a sector or cluster etc.) occupied by a plurality of files (col. 11 line 56-65) of the file system (drawing references 102, 104, and col. 5 line 46) to a locally-stored image file (e.g. an “in-partition image”; col. 5 line 7-10, col. 12 line 65-col. 13 line 5), wherein the locally-stored image file (e.g. an “in-partition image”; col. 5 line 7-10, col. 12 line 65-col. 13 line 5) is located within (drawing reference 420) the same partition (col. 5 lines 7-8, and line 52) as the file system (102, 104, and col. 5 line 46) being backed up (col. 5 line 40-45, col. 14 line 22-23, col. 19 line 6-8); and

adding a directory map (col. 10 line 9-col. 11 line 2 and col. 19 line 20-32) to the locally-stored image file (e.g. an “in-partition image”; col. 5 line 7-10, col. 12 line 65-col. 13 line 5) that associates copied allocation units (col. 10 line 38) in the locally-stored image file (e.g. an “in-partition image”; col. 5 line 7-10, col. 12 line 65-col. 13 line 5) with names of corresponding files (col. 10 line 51-54) from the file system (102, 104, and col. 5 line 46); and

subsequent to creating the locally-stored image file, protecting the locally-stored image file from accidental user deletion or modification (col. 15 line 7-10, col. 20 lines 10-12).

Although Jenevein teaches a desire to protect an image file (e.g. col. 20 line 10-12), Jenevein does not appear to teach protection by initiating a process at system startup that opens the locally-stored image file to block subsequent processes from accessing the locally-stored image file.

Gafken, however, teaches initiating a process at system startup (col. 3 lines 24-31 and col. 12 lines 8-11) that opens (fig. 5; e.g. numeral 525) the locally-stored image file to block subsequent processes from accessing the locally-stored image file (col. 3 lines 29-31, col. 12

lines 8-11) for proving a locking mechanism that operates during start-up to protect system critical information stored in blocks from undesired alterations (Gafken, col. 10, lines 30-33).

Accordingly, in the same field of endeavor, (i.e. protection from inadvertent/accidental modifications/alterations and protecting images), it would have been obvious to one of ordinary skill in the data processing art at the time of the present invention to combine the teachings of the cited references because the teachings of Gafken would have given Jenevein optimized image protection for the benefit of increasing consistency and integrity of the image itself (as needed by Jenevein, col. 5 lines 62-63). Ultimately, Gafken would have provided an improved system to protect system critical information stored in blocks from undesired alterations (Gafken, col. 10, lines 30-33 as needed by Jenevein, col. 20 lines 10-12).

**With Respect to claim 2.** Jenevein teaches the method of claim 1, wherein copying comprises compressing at least a subset of the allocation units (col. 8 line 63-64).

**With Respect to claim 3.** Jenevein teaches the method of claim 1, wherein copying comprises: maintaining a record of a pre-imaging state of the file system (col. 5 line 58-59); and copying only allocation units occupied by files included within the pre-imaging state of the file system (col. 5 line 60-67).

**With Respect to claim 4.** Jenevein teaches the method of claim I, wherein adding comprises grouping within the locally-stored image file the copied allocation units for individual files of the file system (col. 13 line 42-47).

**With Respect to claim 5.** Jenevein teaches The method of claim 1, wherein copying comprises storing within the locally-stored image file one or more attributes related to each file, wherein the attributes comprise at least one of ownership attributes, access-control attributes, timestamp attributes, archival attributes, indexing attributes, encryption attributes, and compression attributes (col. 10 line 49; e.g. the image comprises a date/time of creation).

**With Respect to claim 6.** Jenevein teaches The method of claim 1, further comprising marking a beginning point (col. 11 line 2, col. 14 line 15) of the locally-stored image file to assist in locating the locally-stored image file (col. 14 line 29-48) in the event of directory area corruption (col. 11 line 38-42).

**With Respect to claim 7.** Jenevein teaches The method of claim 6, wherein marking comprises storing a unique beginning-of-image marker at an initial allocation unit occupied by the locally-stored image file (col. 14 line 15-16).

**With Respect to claim 8.** Jenevein teaches The method of claim 6, wherein marking comprises storing at a predetermined area of the partition a location of an initial allocation unit occupied by the locally-stored image file (col. 5 line 21-24).

**With Respect to claim 11.** Jenevein teaches A method for restoring a file system to a partition comprising a plurality of allocation units, the method comprising:

accessing (col. 14 line 28; e.g. locating an image) a locally-stored image file (e.g. an “in-partition image”; col. 5 line 7-10, col. 12 line 65-col. 13 line 5) located within (drawing reference 420) the partition (col. 5 lines 7-8, and line 52) to which the file system (drawing references 102, 104, and col. 5 line 46) is to be restored (col. 5 line 40-45, col. 14 line 22-23, col. 19 line 6-8), the locally-stored image file (e.g. an “in-partition image”; col. 5 line 7-10, col. 12 line 65-col. 13 line 5) comprising a directory map (col. 10 line 9-col. 11 line 2 and col. 19 line 20-32) and file data for a plurality of files (col. 10 line 36-45);

initializing at least a subset (col. 1 line 41-46; e.g. formatting a partition) of the allocation units (col. 3 line 52-53, col. 5 line 31; e.g. a sector or cluster etc.) of the partition not occupied by the locally-stored image file (e.g. an “in-partition image”; col. 5 line 7-10, col. 12 line 65-col. 13 line 5) including one or more allocation units (col. 3 line 52-53, col. 5 line 31; e.g. a sector or cluster etc.) used for a directory area of the partition;

extracting the file data from the locally-stored image file (e.g. an “in-partition image”; col. 5 line 7-10, col. 12 line 65-col. 13 line 5) into the initialized allocation units without disturbing the locally-stored image file (abstract, col. 22 line 16-21); and

creating a new directory area for the partition (col. 20 line 50-51) using the directory map (col. 10 line 9-col. 11 line 2 and col. 19 line 20-32; e.g. the use of an image for restoration describes creating a new directory in the partition being restored).

protecting the locally-stored image file from accidental user deletion or modification subsequent to creation of the locally-stored image file (col. 15 line 7-10).



Although Jenevein teaches a desire to protect an image file (e.g. col. 20 line 10-12), Jenevein does not appear to teach initiating a process at system startup that opens the locally-stored image file to block subsequent processes from accessing the locally-stored image file.

Gafken, however, teaches initiating a process at system startup (col. 3 lines 24-31 and col. 12 lines 8-11) that opens (fig. 5; e.g. numeral 525) the locally-stored image file to block subsequent processes from accessing the locally-stored image file (col. 3 lines 29-31, col. 12 lines 8-11) for providing a locking mechanism that operates during start-up to protect system critical information stored in blocks from undesired alterations (Gafken, col. 10, lines 30-33).

Accordingly, in the same field of endeavor, (i.e. protection from inadvertent/accidental modifications/alterations and protecting images), it would have been obvious to one of ordinary skill in the data processing art at the time of the present invention to combine the teachings of the cited references because the teachings of Gafken would have given Jenevein optimized image protection for the benefit of increasing consistency and integrity of the image itself (as needed by Jenevein, col. 5 lines 62-63). Ultimately, Gafken would have provided an improved system to protect system critical information stored in blocks from undesired alterations (Gafken, col. 10, lines 30-33 as needed by Jenevein, col. 20 lines 10-12)

**With Respect to claim 12.** Jenevein teaches The method of claim 11, wherein the directory map associates names for the plurality of files with corresponding portions of the file data (col. 10 line 50-60), and wherein creating comprises generating a new directory area for the partition that associates the file names with the extracted file data (col. 1 line 41-43, col. 7 line 44-50).

**With Respect to claim 13.** Jenevein teaches The method of claim 11, wherein creating comprises adding an indication of the locally-stored image file to the new directory area (col. 9 line 10-15).

**With Respect to claim 14.** Jenevein teaches The method of claim 11, wherein extracting comprises decompressing at least a subset of the file data (col. 12 line 46-47).

**With Respect to claim 15.** Jenevein teaches The method of claim 11, wherein the directory map indicates at least one attribute for a file (col. 10 line 37-42), and wherein creating comprises setting the at least one attribute for the file in the directory area (col. 10 line 25-67), wherein the at least one attribute is comprises at least one of an ownership attribute, an access control attribute, a timestamp attribute, an archival attribute, an indexing attribute, an encryption attribute, and a compression attribute (col. 10 line 49; e.g. the image comprises a date/time of creation).

**With Respect to claim 16.** Jenevein teaches The method of claim 11, wherein accessing comprises searching for an allocation unit containing a unique beginning-of-image marker (col. 14 line 15-16) for the locally-stored image file (col. 14 line 29).

**With Respect to claim 17.** Jenevein teaches The method of claim 11, wherein accessing comprises reading from a predetermined area of the partition a location of an initial allocation unit of the locally-stored image file (col. 5 line 21-24).

**With Respect to claim 18.** Jenevein teaches The method of claim 11, further comprising defragmenting the locally-stored image file within the partition prior to extracting the file data (col. 14 line 36-37).

**With respect to claim 21.** Jenevein teaches An apparatus for backing up a file system in a partition comprising a plurality of allocation units, the apparatus comprising:

a processor (602);

a local imager (618) programmed to create a locally-stored image file (step 704) by copying each allocation unit (col. 3 line 52-53, col. 5 line 31; e.g. a sector or cluster etc.) occupied by a plurality of files (col. 11 line 56-65) of the file system (drawing references 102, 104, and col. 5 line 46) to the locally-stored image file (e.g. an “in-partition image”; col. 5 line 7-10, col. 12 line 65-col. 13 line 5),

wherein the locally-stored image file (e.g. an “in-partition image”; col. 5 line 7-10, col. 12 line 65-col. 13 line 5) is located within (drawing reference 420) the same partition (col. 5 lines 7-8, and line 52) as the file system (102, 104, and col. 5 line 46) being backed up (col. 5 line 40-45, col. 14 line 22-23, col. 19 line 6-8); and

wherein the local imager (618) is to add a directory map col. 10 line 9-col. 11 line 2 and col. 19 line 20-32) to the locally-stored image file that associates copied allocation units (col. 10

line 38) in the locally-stored image file (e.g. an “in-partition image”; col. 5 line 7-10, col. 12 line 65-col. 13 line 5) with names of corresponding files (col. 10 line 51-54) from the file system (102, 104, and col. 5 line 46); and

a protection component (col. 20 line 6; PowerQuest Drive image) programmed to protect the locally-stored image file from accidental user deletion or modification subsequent to creation of the locally-stored image file (col. 15 line 7-10, col. 20 lines 10-12).

Although Jenevein teaches a desire to protect an image file (e.g. col. 20 line 10-12), Jenevein does not appear to teach initiating a process at system startup that opens the locally-stored image file to block subsequent processes from accessing the locally-stored image file.

Gafken, however, teaches initiating a process at system startup (col. 3 lines 24-31 and col. 12 lines 8-11) that opens (fig. 5; e.g. numeral 525) the locally-stored image file to block subsequent processes from accessing the locally-stored image file (col. 3 lines 29-31, col. 12 lines 8-11) for providing a locking mechanism that operates during start-up to protect system critical information stored in blocks from undesired alterations (Gafken, col. 10, lines 30-33).

Accordingly, in the same field of endeavor, (i.e. protection from inadvertent/accidental modifications/alterations and protecting images), it would have been obvious to one of ordinary skill in the data processing art at the time of the present invention to combine the teachings of the cited references because the teachings of Gafken would have given Jenevein optimized image protection for the benefit of increasing consistency and integrity of the image itself (as needed by Jenevein, col. 5 lines 62-63). Ultimately, Gafken would have provided an improved system to protect system critical information stored in blocks from undesired alterations (Gafken, col. 10, lines 30-33 as needed by Jenevein, col. 20 lines 10-12).

**With Respect to claim 22.** Jenevein teaches The apparatus of claim 21, wherein the local imager is configured to compress at least a subset of the allocation units copied to the locally-stored image file (col. 8 line 63-64).

**With Respect to claim 23.** Jenevein teaches The apparatus of claim 21, wherein the local imager is configured to maintain a record of a pre-imaging state of the file system (col. 5 line 58-59) and to copy only allocation units occupied by files included within the pre-imaging state of the file system (col. 5 line 60-67).

**With Respect to claim 24.** Jenevein teaches The apparatus of claim 21, wherein the local imager is configured to group within the locally-stored image file the copied allocation units for individual files of the file system (col. 13 line 42-47).

**With Respect to claim 25.** Jenevein teaches The apparatus of claim 21, wherein the local imager is configured to store within the locally-stored image file one or more attributes relating to at least one file of the file system, wherein the file attributes are selected from the group consisting of ownership attributes, access-control attributes, timestamp attributes, archival attributes, indexing attributes, encryption attributes, and compression attributes (col. 10 line 49; e.g. the image comprises a date/time of creation)..

**With Respect to claim 26.** Jenevein teaches The apparatus of claim 21, wherein the local imager is configured to mark a beginning point of the locally-stored image file to assist in locating the locally-stored image file in the event of directory area corruption (col. 11 line 38-42).

**With Respect to claim 27.** Jenevein teaches The apparatus of claim 26, wherein the local imager is configured to mark the beginning point by storing a unique beginning-of-image marker (col. 14 line 15-16) at an initial allocation unit occupied by the locally-stored image file (col. 14 line 29).

**With Respect to claim 28.** Jenevein teaches The apparatus of claim 26, wherein the local imager is configured to mark the beginning point by storing at a predetermined area of the partition a location of an initial allocation unit occupied by the locally-stored image file (col. 5 line 21-24).

**With Respect to claim 31.** Jenevein teaches An apparatus for restoring a file system to a partition comprising a plurality of allocation units, the apparatus comprising:

a processor (602);

an image locator (620) to find (col. 14 line 29- line 48) a locally-stored image file (e.g. an “in-partition image”) located within (420) the partition (col. 5 lines 7-8, and line 52) to which the file system is to be restored (col. 5 line 40-45, col. 14 line 22-23, col. 19 line 6-8), the locally-

stored image file (e.g. an “in-partition image”) comprising a directory map (col. 10 line 9-col. 11 line 2 and col. 19 line 20-32) and file data for a plurality of files (col. 10 line 50-67);

a media formatter (602, col. 1 line 41-45) to initialize (col. 1 line 41-46; e.g. formatting a partition) at least a subset of the allocation units (col. 3 line 52-53, col. 5 line 31; e.g. a sector or cluster etc.) of the partition (col. 5 lines 7-8, and line 52) not occupied by the locally-stored image file (e.g. an “in-partition image”) including one or more allocation units used for a directory area (col. 20 line 50-51) of the partition (col. 5 lines 7-8, and line 52);

a data extractor (734) to extract the file data from the locally-stored image file into the initialized allocation units without disturbing the locally-stored image file (e.g. an “in-partition image”); and

a directory area builder (712) to build a new directory area (col. 20 line 50-51) for the partition using the directory map (col. 10 line 9-col. 11 line 2 and col. 19 line 20-32); and

a protection component programmed to protect the locally-stored image file from accidental user deletion or modification subsequent to creation of the locally-stored image file (col. 15 line 7-10, col. 20 lines 10-12).

Although Jenevein teaches a desire to protect an image file (e.g. col. 20 line 10-12), Jenevein does not appear to teach initiating a process at system startup that opens the locally-stored image file to block subsequent processes from accessing the locally-stored image file.

Gafken, however, teaches initiating a process at system startup (col. 3 lines 24-31 and col. 12 lines 8-11) that opens (fig. 5; e.g. numeral 525) the locally-stored image file to block subsequent processes from accessing the locally-stored image file (col. 3 lines 29-31, col. 12

lines 8-11) for proving a locking mechanism that operates during start-up to protect system critical information stored in blocks from undesired alterations (Gafken, col. 10, lines 30-33).

Accordingly, in the same field of endeavor, (i.e. protection from inadvertent/accidental modifications/alterations and protecting images), it would have been obvious to one of ordinary skill in the data processing art at the time of the present invention to combine the teachings of the cited references because the teachings of Gafken would have given Jenevein optimized image protection for the benefit of increasing consistency and integrity of the image itself (as needed by Jenevein, col. 5 lines 62-63). Ultimately, Gafken would have provided an improved system to protect system critical information stored in blocks from undesired alterations (Gafken, col. 10, lines 30-33 as needed by Jenevein, col. 20 lines 10-12).

**With Respect to claim 32.** Jenevein teaches The apparatus of claim 31, wherein the directory map associates names for the plurality of files with corresponding portions of the file data, and wherein the directory area builder is configured to generate a new directory area for the partition that associates the file names with the extracted file data (col. 1 line 41-43, col. 7 line 44-50).

**With Respect to claim 33.** Jenevein teaches The apparatus of claim 31, wherein the directory area builder is configured to add an indication of the locally-stored image file to the new directory area (col. 9 line 10-15).



**With Respect to claim 34.** Jenevein teaches The apparatus of claim 31, wherein the data extractor is configured to decompress at least a subset of the file data (col. 12 line 46-47).

**With Respect to claim 35.** Jenevein teaches The apparatus of claim 31, wherein the directory map indicates at least one attribute for a file, wherein the directory area builder is to set the at least one attribute of the file in the directory area, and wherein the at least one attribute comprises at least one of an ownership attribute, an access control attribute, a timestamp attribute, an archival attribute, an indexing attribute, an encryption attribute, and a compression attribute (col. 10 line 49; e.g. the image comprises a date/time of creation).

**With Respect to claim 36.** Jenevein teaches The apparatus of claim 31, wherein the image locator is configured to search for an allocation unit containing a unique beginning-of-image marker (col. 14 line 15-16) for the locally-stored image file (col. 14 line 29).

**With Respect to claim 37.** Jenevein teaches The method of claim 31, wherein the image locator is configured to read from a predetermined area of the partition a location of at least a first allocation unit of the locally-stored image file (col. 5 line 21-24).

**With Respect to claim 38.** Jenevein teaches The apparatus of claim 31, further comprising an image defragmenter to defragment the locally-stored image file within the partition before the data extractor extracts the file data (col. 14 line 36-37).

**With Respect to claim 41.** Jenevein teaches A method for localized backup and restoration of a file system in a partition comprising a plurality of allocation units, the method comprising:

creating a locally-stored image file (step 704) by copying (col. 5 line 31-32) each allocation unit (col. 3 line 52-53, col. 5 line 31; e.g. a sector or cluster etc.) occupied by a plurality of files (col. 11 line 56-65) of the file system (drawing references 102, 104, and col. 5 line 46) to a locally-stored image file (e.g. an “in-partition image”; col. 5 line 7-10, col. 12 line 65-col. 13 line 5), wherein the locally-stored image file (e.g. an “in-partition image”; col. 5 line 7-10, col. 12 line 65-col. 13 line 5) is located within (drawing reference 420) the same partition (col. 5 lines 7-8, and line 52) as the file system (102, 104, and col. 5 line 46) being backed up (col. 5 line 40-45, col. 14 line 22-23, col. 19 line 6-8); and

adding a directory map (col. 10 line 9-col. 11 line 2 and col. 19 line 20-32) to the locally-stored image file (e.g. an “in-partition image”; col. 5 line 7-10, col. 12 line 65-col. 13 line 5) that associates copied allocation units (col. 10 line 38) in the locally-stored image file (e.g. an “in-partition image”; col. 5 line 7-10, col. 12 line 65-col. 13 line 5) with names of corresponding files (col. 10 line 51-54) from the file system (102, 104, and col. 5 line 46)

locating the locally-stored image file within the partition (col. 14 line 29-48);

initializing at least a subset (col. 1 line 41-46; e.g. formatting a partition) of the allocation units (col. 3 line 52-53, col. 5 line 31; e.g. a sector or cluster etc.) of the partition not occupied by the locally-stored image file (e.g. an “in-partition image”; col. 5 line 7-10, col. 12 line 65-col. 13 line 5) including one or more allocation units (col. 3 line 52-53, col. 5 line 31; e.g. a sector or cluster etc.) used for a directory area of the partition (col. 5 lines 7-8, and line 52);

extracting file data from the locally-stored image file into the initialized allocation units without disturbing the locally-stored image file (abstract, col. 22 line 16-21); and

creating a new directory area for the partition (col. 20 line 50-51) using the directory map (col. 10 line 9-col. 11 line 2 and col. 19 line 20-32); and

subsequent to creating the locally-stored image file, protecting the locally-stored image file from accidental user deletion or modification (col. 15 line 7-10, col. 2 lines 10-12).

Although Jenevein teaches a desire to protect an image file (e.g. col. 20 line 10-12), Jenevein does not appear to teach initiating a process at system startup that opens the locally-stored image file to block subsequent processes from accessing the locally-stored image file.

Gafken, however, teaches initiating a process at system startup (col. 3 lines 24-31 and col. 12 lines 8-11) that opens (fig. 5; e.g. numeral 525) the locally-stored image file to block subsequent processes from accessing the locally-stored image file (col. 3 lines 29-31, col. 12 lines 8-11) for providing a locking mechanism that operates during start-up to protect system critical information stored in blocks from undesired alterations (Gafken, col. 10, lines 30-33).

Accordingly, in the same field of endeavor, (i.e. protection from inadvertent/accidental modifications/alterations and protecting images), it would have been obvious to one of ordinary skill in the data processing art at the time of the present invention to combine the teachings of the cited references because the teachings of Gafken would have given Jenevein optimized image protection for the benefit of increasing consistency and integrity of the image itself (as needed by Jenevein, col. 5 lines 62-63). Ultimately, Gafken would have provided an improved system to protect system critical information stored in blocks from undesired alterations (Gafken, col. 10, lines 30-33 as needed by Jenevein, col. 20 lines 10-12).

**With Respect to claim 42.** Jenevein teaches A computer-readable storage medium comprising program code for backing up a file system in a partition comprising a plurality of allocation units, the computer-readable storage medium comprising:

program code for creating a locally-stored image file (step 704) by copying (col. 5 line 31-32) each allocation unit (col. 3 line 52-53, col. 5 line 31; e.g. a sector or cluster etc.) occupied by a plurality of files (col. 11 line 56-65) of the file system (drawing references 102, 104, and col. 5 line 46) to a locally-stored image file (e.g. an “in-partition image”; col. 5 line 7-10, col. 12 line 65-col. 13 line 5), wherein the locally-stored image file (e.g. an “in-partition image”; col. 5 line 7-10, col. 12 line 65-col. 13 line 5) is located within (drawing reference 420) the same partition (col. 5 lines 7-8, and line 52) as the file system (102, 104, and col. 5 line 46) being backed up (col. 5 line 40-45, col. 14 line 22-23, col. 19 line 6-8); and

adding a directory map (col. 10 line 9-col. 11 line 2 and col. 19 line 20-32) to the locally-stored image file (e.g. an “in-partition image”; col. 5 line 7-10, col. 12 line 65-col. 13 line 5) that associates copied allocation units (col. 10 line 38) in the locally-stored image file (e.g. an “in-partition image”; col. 5 line 7-10, col. 12 line 65-col. 13 line 5) with names of corresponding files (col. 10 line 51-54) from the file system (102, 104, and col. 5 line 46); and

program code for protecting the locally-stored image file from accidental user deletion or modification subsequent to creation of the locally stored image file (col. 15 line 7-10, col. 20 lines 10-12).

Although Jenevein teaches a desire to protect an image file (e.g. col. 20 line 10-12), Jenevein does not appear to teach initiating a process at system startup that opens the locally-stored image file to block subsequent processes from accessing the locally-stored image file.

Gafken, however, teaches initiating a process at system startup (col. 3 lines 24-31 and col. 12 lines 8-11) that opens (fig. 5; e.g. numeral 525) the locally-stored image file to block subsequent processes from accessing the locally-stored image file (col. 3 lines 29-31, col. 12 lines 8-11) for providing a locking mechanism that operates during start-up to protect system critical information stored in blocks from undesired alterations (Gafken, col. 10, lines 30-33).

Accordingly, in the same field of endeavor, (i.e. protection from inadvertent/accidental modifications/alterations and protecting images), it would have been obvious to one of ordinary skill in the data processing art at the time of the present invention to combine the teachings of the cited references because the teachings of Gafken would have given Jenevein optimized image protection for the benefit of increasing consistency and integrity of the image itself (as needed by Jenevein, col. 5 lines 62-63). Ultimately, Gafken would have provided an improved system to protect system critical information stored in blocks from undesired alterations (Gafken, col. 10, lines 30-33 as needed by Jenevein, col. 20 lines 10-12).

**With Respect to claim 43.** Jenevein teaches A computer-readable storage medium comprising program code for restoring a file system to a partition comprising a plurality of allocation units, the computer-readable storage medium comprising:

program code to access (col. 14 line 28; e.g. locating an image) a locally-stored image file (e.g. an "in-partition image"; col. 5 line 7-10, col. 12 line 65-col. 13 line 5) located within

(drawing reference 420) the partition (col. 5 lines 7-8, and line 52) to which the file system (drawing references 102, 104, and col. 5 line 46) is to be restored (col. 5 line 40-45, col. 14 line 22-23, col. 19 line 6-8), the locally-stored image file (e.g. an “in-partition image”; col. 5 line 7-10, col. 12 line 65-col. 13 line 5) comprising a directory map (col. 10 line 9-col. 11 line 2 and col. 19 line 20-32) and file data for a plurality of files (col. 10 line 36-45);

program code to initialize at least a subset (col. 1 line 41-46; e.g. formatting a partition) of the allocation units (col. 3 line 52-53, col. 5 line 31; e.g. a sector or cluster etc.) of the partition not occupied by the locally-stored image file (e.g. an “in-partition image”; col. 5 line 7-10, col. 12 line 65-col. 13 line 5) including one or more allocation units (col. 3 line 52-53, col. 5 line 31; e.g. a sector or cluster etc.) used for a directory area of the partition;

program code to extract the file data from the locally-stored image file into the initialized allocation units without disturbing the locally-stored image file (abstract, col. 22 line 16-21); and

program code to create a new directory area for the partition using the directory map (col. 10 line 9-col. 11 line 2 and col. 19 line 20-32); and

program code to protect the locally-stored image file from accidental user deletion or modification subsequent to creation of the locally-stored image file (col. 15 line 7-10, col. 20 lines 10-12).

Although Jenevein teaches a desire to protect an image file (e.g. col. 20 line 10-12), Jenevein does not appear to teach initiating a process at system startup that opens the locally-stored image file to block subsequent processes from accessing the locally-stored image file.

Gafken, however, teaches initiating a process at system startup (col. 3 lines 24-31 and col. 12 lines 8-11) that opens (fig. 5; e.g. numeral 525) the locally-stored image file to block

subsequent processes from accessing the locally-stored image file (col. 3 lines 29-31, col. 12 lines 8-11) for providing a locking mechanism that operates during start-up to protect system critical information stored in blocks from undesired alterations (Gafken, col. 10, lines 30-33).

Accordingly, in the same field of endeavor, (i.e. protection from inadvertent/accidental modifications/alterations and protecting images), it would have been obvious to one of ordinary skill in the data processing art at the time of the present invention to combine the teachings of the cited references because the teachings of Gafken would have given Jenevein optimized image protection for the benefit of increasing consistency and integrity of the image itself (as needed by Jenevein, col. 5 lines 62-63). Ultimately, Gafken would have provided an improved system to protect system critical information stored in blocks from undesired alterations (Gafken, col. 10, lines 30-33 as needed by Jenevein, col. 20 lines 10-12).

**Claims 10, 20, 30, and 40 are rejected under 35 U.S.C. 103(a) as being unpatentable over Jenevein and Gafken and further in view of GB Patent No. GB2376093 issued to Leech, Guy ('Leech' hereafter, see supplied printout pages 1-6 published 2002-12-04).**

**With Respect to claim 10.** Jenevein does not explicitly recite wherein protecting the locally-stored image file further comprises providing a filter driver that intercepts and denies requests to access the locally-stored image file.

Leech, however, explicitly recites protecting by providing a filter driver that intercepts (page 5, last 7 paragraphs; "the filter driver intercepts the execution requests 22 before the lower level file system drivers...") and denies requests to access (abstract and page 4; "the rule may deny requests") for protecting data.

Accordingly, in the same field of endeavor, (i.e. data protection), it would have been obvious to one of ordinary skill in the data processing art at the time of the present invention to combine the teachings of the cited references because the teachings of Leech would have given Jenevein a way to protect locally-stored image files from so they are not easily overwritten or deleted (the need being disclosed in col. 15 lines 7-10 and col. 20 line 10-12).

**With respect to claim 20.** Jenevein does not explicitly recite wherein protecting the locally-stored image file further comprises providing a filter driver that intercepts and denies requests to access the locally-stored image file.

Leech, however, explicitly recites protecting by providing a filter driver that intercepts (page 5, last 7 paragraphs; “the filter driver intercepts the execution requests 22 before the lower level file system drivers...”) and denies requests to access (abstract and page 4; “the rule may deny requests”) for protecting data.

Accordingly, in the same field of endeavor, (i.e. data protection), it would have been obvious to one of ordinary skill in the data processing art at the time of the present invention to combine the teachings of the cited references because the teachings of Leech would have given Jenevein a way to protect locally-stored image files from so they are not easily overwritten or deleted (the need being disclosed in col. 15 lines 7-10 and col. 20 line 10-12).

**With Respect to claim 30.** Jenevein does not explicitly recite wherein protecting the locally-stored image file further comprises providing a filter driver that intercepts and denies requests to access the locally-stored image file.



Leech, however, explicitly recites protecting by providing a filter driver that intercepts (page 5, last 7 paragraphs; “the filter driver intercepts the execution requests 22 before the lower level file system drivers...”) and denies requests to access (abstract and page 4; “the rule may deny requests”) for protecting data.

Accordingly, in the same field of endeavor, (i.e. data protection), it would have been obvious to one of ordinary skill in the data processing art at the time of the present invention to combine the teachings of the cited references because the teachings of Leech would have given Jenevein a way to protect locally-stored image files from so they are not easily overwritten or deleted (the need being disclosed in col. 15 lines 7-10 and col. 20 line 10-12).

**With Respect to claim 40.** Jenevein does not explicitly recite wherein protecting the locally-stored image file further comprises providing a filter driver that intercepts and denies requests to access the locally-stored image file.

Leech, however, explicitly recites protecting by providing a filter driver that intercepts (page 5, last 7 paragraphs; “the filter driver intercepts the execution requests 22 before the lower level file system drivers...”) and denies requests to access (abstract and page 4; “the rule may deny requests”) for protecting data.

Accordingly, in the same field of endeavor, (i.e. data protection), it would have been obvious to one of ordinary skill in the data processing art at the time of the present invention to combine the teachings of the cited references because the teachings of Leech would have given Jenevein a way to protect locally-stored image files from so they are not easily overwritten or deleted (the need being disclosed in col. 15 lines 7-10 and col. 20 line 10-12).

**(10) Response to Argument**

On page 7 in the arguments section of the Appeal Brief (herein ‘Brief’), Appellant argues that the cited references fail to teach the protection element features of 1, 11, 21, 31, 41, 42, and 43. Specifically, Appellant argues at the bottom of page 8 throughout to page 11 in the Brief that Gafken (who was relied upon for teaching this feature) does not teach of protecting a locally-stored image file by initiating a process at system startup that opens the locally-stored image file to block subsequent processes from accessing the locally-stored image file. As detailed below, Examiner respectfully disagrees:

As indicated in the rejection to claims 1, 11, 21, 31, 41, 42, and 43, Jenevein was relied upon to teach the claimed locally-stored image file. Jenevein, however, did not appear to teach the protection feature as specifically claimed. Therein, Gafken was relied upon to teach this deficiency.

As relied upon in the rejection to claims 1, 11, 21, 31, 41, 42, and 43, Gafken teaches this protection feature at least in col. 3, lines 24-31. Herein, Gafken teaches start-up code that locks and/or locks down any other blocks that are desired to be locked and/or locked down based on the update so that read and/or write and erase operations are prevented before other software is able to run. Gafken further teaches in col. 12, lines 8-11 that while a system start-up routine is executing, viruses, system software, and/or other processor commands cannot alter the information stored in the memory array 130.

Accordingly, in light of the above, Gafken teaches that while start-up code is executing and in control (i.e. “during system startup”), outside access and alterations are prevented.

Examiner also notes that updates to code [images]<sup>1</sup> are performed while the aforementioned start-up code has control of the system (e.g. col. 14, lines 6-8) in order to prevent alterations or malicious destruction (col. 14, lines 8-9). Furthermore, Gafken also teaches that updates can be made to code and/or data in locked down blocks while protecting the information stored in the locked down blocks from being otherwise altered. This happens while the start-up routine has exclusive control of the system (col. 14, lines 27-35).

Thus in light of the above, Gafken teaches the control and prevention of alterations to specific information, and more particularly code images while start-up code has control of the system (i.e. performing a lock-down and update process at system startup). Moreover, since updates are made to this information during the lock-down period at start-up, it can be interpreted that this information is opened in order to perform the updates.

Further, on pages 11-12 of the Brief, Appellant argues that Gafken does not teach protecting a locally-stored image file and further submits that a “code image” is not analogous to the locally stored image file. Examiner respectfully disagrees and submits that Gafken provides a protection mechanism as disclosed above which is applicable to images, and more generally, to data. As Jenevein teaches the locally-stored image file as claimed, Gafken teaches a protection mechanism for data. As such (in the same field of endeavor – protection of images), Jenevein would have been improved by the benefits listed above in the motivation for combination.

---

<sup>1</sup> Gafken refers to the code as a code image (e.g. col. 13, line 31)

Accordingly, in light of Jenevein's need for image protection (col. 5 lines 62-63), it would have been obvious to combine Gafken's protection feature that updates (i.e. opens) information, or images, during system start-up to block subsequent accessing processes for the benefits of optimized image protection and prevention of alterations as provided in the rejection to the foregoing claims.

**(11) Related Proceeding(s) Appendix**

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/ROBERT TIMBLIN/

Primary Examiner, Art Unit 2167

Conferees:

/John R. Cottingham/

Supervisory Patent Examiner, Art Unit 2167

/CHARLES KIM/

Supervisory Patent Examiner, Art Unit 2157